

Amendments to the Claims:

1. (Currently Amended) An apparatus for smoothly playing a pre-determined sequence of songs transmitted from a server over the Internet, the apparatus comprising;

a processor;

a first memory that stores at least one control program used by said by the processor to control the playing of the predetermined sequence of songs, said program the at least one control program including computer-readable instructions for specifying specifying a number of beginning portions of songs to cache in advance and size of a pre-buffer cache; and

a second memory which is available to said ~~at the at least one~~ control program for operations, wherein said ~~at the at least one~~ control program causes said processor ~~the processor~~ at least to:

as ~~soon as a song starts to play, start to download, consecutively, a first small beginning~~ portion of each of a number of songs which are, in the pre-determined sequence, subsequent to the playing song, in playing, said downloaded small portions being pre-cached in a to the pre-buffer cache, wherein the pre-buffer cache which is an area in said of the second memory;

as ~~soon as the user skips if playback is skipped~~ to a target song whose first small for which the beginning portion has been pre-cached downloaded to the pre-buffer cache, start to play the downloaded beginning first small portion of said target the target song; and

~~at the same timewhile playing the downloaded beginning portion of the target song, start to download the rest of said target the target song so that as soon as the playing of the first small portion of said target song ends, start to play the rest of said target song which is being downloaded from the server over the Internet.~~

2. (Currently Amended) The apparatus of claim 1, wherein ~~said first-small-the~~ beginning portion of the target song is approximately the data required for playing of the first ten seconds of the target song.

3. (Currently Amended) The apparatus of claim 1, wherein ~~said-number-the~~ number of beginning portions of songs to cache in advance is five.

4. (Currently Amended) The apparatus of claim 1, wherein ~~said-number-the~~ number of beginning portions of songs to cache in advance is all songs in the pre-determined sequence of songs that are subsequent to the playing song-in-playing.

5. (Currently Amended) The apparatus of claim 1, wherein ~~said-buffer-the~~ pre-buffer cache follows a first-in first-out algorithm and allows writing while reading.

6. (Currently Amended) A method for smoothly playing a pre-determined sequence of songs transmitted from a remote server to a local playback device over the Internet, comprising the steps of:

(a) ~~as soon as a~~ song starts to play on the local playback device,
downloading, consecutively, a ~~first-small-beginning~~ portion of each of a
number of songs which are, in the pre-determined sequence, subsequent
to ~~said-song-in-playing~~ the playing song; and

(b) ~~pre-caching said-downloaded-small-the downloaded~~ beginning portions in
a ~~buffer which is an area of said-local device's to a~~ pre-buffer cache of a
memory of the local playback device, wherein the number of beginning
portions of songs to pre-cache in advance and size of a pre-buffer-the pre-
buffer cache can be specified- are specified by a function call.

7. (Currently Amended) The method of claim 6, further comprising the steps
of:

(c) ~~as soon as the user skips if~~ playback is skipped from a playing song-in
playing to a target song, checking whether a ~~file for said-the~~ beginning

- ~~portion of the target song exists in said buffer, wherein if the check result is yes, continuing on step (d) is in the pre-buffer cache; and~~
- (d) ~~if the beginning portion of the target song is in the pre-buffer cache, playing the first small beginning portion of said target the target song from the pre-buffer cache; and~~
- (e) ~~as soon as step (d) starts, downloading the rest of said at least a portion of the target song which is not in the pre-buffer cache, and ;~~
- (f) ~~as soon as step (d) starts, deleting any pre-cached beginning portions of any songs prior to said target the target song in said in the pre-determined sequence from the pre-buffer cache; and~~
- (g) ~~playing the rest of said target song which is being downloaded from the server over the Internet.~~

8. (Currently Amended) The method of claim 7, further comprising the step
of:

- (h) ~~as soon as step (d) starts, continuing on step (a), if the beginning portion of the target song is in the pre-buffer cache, downloading, consecutively, a beginning portion of each of a number of songs which are, in the pre-determined sequence, subsequent to the target song, wherein if beginning portions of the one or more songs subsequent to said target the target song are already pre-cached in the pre-buffer cache, skipping said one the downloading of the beginning portions of the one or more songs already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ones, executively songs, consecutively, to make up said number the number of beginning portions of songs to cache in advance.~~

9. (Currently Amended) The method of claim 8, further comprising the steps
of:

- (i) ~~if no skip command is given by the user while said received while the target song is playing, as soon as the playing of said target the target song~~

ends, playing the ~~next song~~ song immediately subsequent to said ~~target~~ the target song; and

- (j) ~~if a skip command is given by the user while said received while the target~~ song is playing, continuing on step (e) checking whether the beginning portion of the song immediately subsequent to the target song is in the pre-buffer cache.

10. (Currently Amended) The method of claim 7, wherein if the check result of step (e) ~~is no beginning portion of the target song is not in the pre-buffer cache, the method further comprises~~ comprising the steps of:

- (k) ~~sending a request to stop transmitting of said the playing song in playing and to start transmitting said target the target song and at least substantially simultaneously;~~
- (l) ~~at the same time with step (k), deleting the beginning portion of pre-cached portion for any song which is prior to said target the target song in the pre-determined sequence of songs from the pre-buffer cache;~~
- (m) ~~downloading said target at least a remaining portion of the target song; and~~
- (n) ~~begin playing said target the target song while being downloaded as soon as said buffer allows so; and a sufficient portion of the target song has been downloaded~~
- (e) ~~at the same time with step (n), continuing on step (a).~~

11. (Currently Amended) The method of claim 10, subsequent to step (n), further comprising the steps of:

- (p) ~~if another skip command is given by the user while said playback is skipped from the target song is playing, continuing on step (e); to another target song, checking whether the beginning portion of the other target song is in the pre-buffer cache; and~~
- (q) ~~if no skip command is given by the user while said playback is not skipped from the target song is playing, as soon as the playing of said target song~~

- ends, ~~playing the first small beginning portion of the next song~~ subsequent to said target the target song after the end of the target song is played and;
- (r) ~~at the same time with step (q), downloading the rest of said at least a~~ portion of the target song which is not in the pre-buffer cache;
- (s) ~~at the same time with step (q), continuing on step (a), wherein if beginning~~ portions of the one or more songs subsequent to said next songs in the pre-determined sequence of songs are already pre-cached in the pre-buffer cache, skipping said one the downloading of the beginning portions of the one or more songs already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ones, executively songs, consecutively, to make up said to the number of beginning portions of songs to cache in advance; and
- (t) ~~subsequent to step (q), playing the rest of the next song which is being~~ download from the server over the Internet

12. (Currently Amended) The method of claim 6, wherein said ~~first small the~~ beginning portion of the target song is approximately the data required ~~for playing of the~~ first ten seconds of the target song.

13. (Currently Amended) The method of claim 6, wherein said ~~number the~~ number of beginning portions of songs to cache in advance is five.

14. (Currently Amended) The method of claim 6, wherein said ~~number the~~ number of beginning portions of songs to cache in advance is all songs in the pre-determined sequence of songs that are subsequent to the playing song in playing.

15. (Currently Amended) The method of claim 6, wherein said ~~buffer the pre-~~ buffer cache follows a first-in first-out algorithm and allows writing while reading.

16. (Currently Amended) A program storage medium readable by a computer, tangibly embodying a program of instructions executable by the computer to perform a method for smoothly playing a pre-determined sequence of songs transmitted from a remote server to a local device the computer over the Internet, comprising the steps of:

- (a) ~~as soon as a song starts to play on the computer,~~ downloading, consecutively, a first small beginning portion of each of a number of songs which are, in the pre-determined sequence, subsequent to said song in playing the playing song; and
- (b) ~~pre-caching said downloaded small~~ the downloaded beginning portions in a buffer which is ~~an area of said local device's~~ to a pre-buffer cache of a memory of the local playback device, wherein the number of beginning portions of songs to pre-cache in advance and size of a pre-buffer ~~the pre-buffer cache can be specified~~ are specified by a function call.

17. (Currently Amended) The program storage medium of claim 16, further comprising the steps of:

- (c) ~~as soon as the user skips if playback is skipped from a playing song in~~ playing to a target song, checking whether a file for said the beginning portion of the target song exists in said buffer, wherein if the check result is yes, continuing on step (d) is in the pre-buffer cache; and
- (d) ~~if the beginning portion of the target song is in the pre-buffer cache,~~ playing the first small beginning portion of said target the target song from the pre-buffer cache; and
- (e) ~~as soon as step (d) starts,~~ downloading the rest of said at least a portion of the target song which is not in the pre-buffer cache, and;
- (f) ~~as soon as step (d) starts,~~ deleting any pre-cached beginning portions of any songs prior to said target ~~the target song~~ in said ~~in the~~ pre-determined sequence from the pre-buffer cache; and
- (g) ~~playing the rest of said target song which is being downloaded from the server over the Internet.~~

18. (Currently Amended) The program storage medium of claim 17, further comprising the step of:

~~(h) as soon as step (d) starts, continuing on step (a), if the beginning portion of the target song is in the pre-buffer cache, downloading a beginning portion of each of a number of songs which are, in the pre-determined sequence, subsequent to the target song, wherein if beginning portions of the one or more songs subsequent to said target the target song are already pre-cached in the pre-buffer cache, skipping said one the downloading of the beginning portions of the one or more songs already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ones, executively songs, consecutively, to make up said number the number of beginning portions of songs to cache in advance.~~

19. (Currently Amended) The program storage medium of claim 18, further comprising the steps of:

~~(i) if no skip command is given by the user while said received while the target song is playing, as soon as the playing of said target the target song ends, playing the next song song immediately subsequent to said target the target song; and~~
~~(j) if a skip command is given by the user while said received while the target song is playing, continuing on step (e) checking whether the beginning portion of the song immediately subsequent to the target song is in the pre-buffer cache.~~

20. (Currently Amended) The program storage medium of claim 17, wherein if the check result of step (c) is no beginning portion of the target song is not in the pre-buffer cache, the method further comprises comprising the steps of:

~~(k) sending a request to stop transmitting of said the playing song in playing and to start transmitting said target the target song, and at least substantially simultaneously;~~

- (l) ~~at the same time with step (k), deleting the beginning portion of pre-cached portion for any song which is prior to said target~~ the target song in the pre-determined sequence of songs; ~~(m) downloading said target from the pre-buffer cache, and downloading at least a remaining portion of the target song; and~~
- (n) ~~begin playing said target~~ the target song while being downloaded as soon as said buffer allows so; and a sufficient portion of the target song has been downloaded
- (o) ~~at the same time with step (n), continuing on step (a).~~

21. (Currently Amended) The program storage medium of claim 20, subsequent to step (n), further comprising the steps of:

- (p) ~~if another skip command is given by the user while said playback is skipped from the target song is playing, continuing on step (c); to another target song, checking whether the beginning portion of the other target song is in the pre-buffer cache; and~~
- (q) ~~if no skip command is given by the user while said playback is not skipped from the target song is playing, as soon as the playing of said target song ends, playing the first small beginning portion of the next song song subsequent to said target~~ the target song after the end of the target song is played;
- (r) ~~at the same time with step (q), downloading the rest of said~~ at least a portion of the target song which is not in the pre-buffer cache;
- (s) ~~at the same time with step (q), continuing on step (a), wherein if beginning portions of the one or more songs subsequent to said next songs in the pre-determined sequence are already pre-cached in the pre-buffer cache, skipping said one~~ the downloading of the beginning portions of one or more songs already having beginning portions in the pre-buffer cache and downloading the beginning portions of the subsequent ones, executively, songs to make up said number the number of beginning portions of songs to cache in advance; and

(t) ~~subsequent to step (q), playing the rest of the next song which is being download from the server over the Internet~~

22. (Currently Amended) The program storage medium of claim 16, wherein ~~said first small~~ the beginning portion of the target song is approximately the data required for playing of the first ten seconds of the target song.

23. (Currently Amended) The program storage medium of claim 16, wherein ~~said number~~ the number of beginning portions of songs to cache in advance is five.

24. (Currently Amended) The program storage medium of claim 16, wherein ~~said number~~ the number of beginning portions of songs to cache in advance is all songs in the pre-determined sequence of songs that are subsequent to the playing song-in playing.

25. (Currently Amended) The program storage medium of claim 16, wherein ~~said buffer~~ the pre-buffer cache follows a first-in first-out algorithm and allows writing while reading.

26. (New) An apparatus for smoothly playing a pre-determined sequence of songs transmitted from a server over the Internet, comprising;

means for controlling the playback of a pre-determined sequence of songs;

means for pre-downloading a beginning portion of a number of songs from the pre-determined sequence of songs; and

means for caching the pre-downloaded beginning portions, wherein the number of beginning portions to be pre-downloaded is configurable via a function call.